

Schließlich gestattet die Regel `rdfs11` die ableitungstechnische Ausnutzung der Transitivität der `rdfs:subClassOf`-Property.

⊗ **Container**

Die folgende Regel weist die `rdfs:member`-Property als Oberproperty aller in der `rdfs:ContainerMembershipProperty`-Klasse enthaltenen Propertys aus.

$$\frac{u \text{ rdf:type rdfs:ContainerMembershipProperty .}}{u \text{ rdfs:subPropertyOf rdfs:member .}} \text{ rdfs12}$$

⊗ **Literale**

Schließlich erlaubt die letzte Regel, über jede als Datentyp identifizierte Resource (die ja mit ihrem Wertebereich identifiziert wird) abzuleiten, dass sie eine Unterklasse aller Literalwerte ist.

$$\frac{u \text{ rdf:type rdfs:Datatype .}}{u \text{ rdfs:subClassOf rdfs:Literal .}} \text{ rdfs13}$$

Bevor wir nun allgemein formulieren können, wie die RDFS-Folgerung durch Ableitungsregeln charakterisiert werden kann, müssen wir noch auf einen Sonderfall eingehen. Ist ein gegebener Graph G inkonsistent (d.h., es gibt keine Interpretation \mathcal{I} , für die $G^{\mathcal{I}} = \text{wahr}$ gilt), so folgt aus ihm *jeder beliebige* Graph (wie man an der modelltheoretischen Semantikdefinition leicht nachvollziehen kann). Im Gegensatz zu anderen Logiken (wie Prädikatenlogik (siehe Anhang A) oder auch OWL, das wir in den nachfolgenden Kapiteln behandeln) gibt es jedoch in RDFS nur sehr eingeschränkte Möglichkeiten, Inkonsistenzen zu erzeugen.

Als Beispiel für eine solche Inkonsistenz betrachten wir die beiden folgenden Tripel:

<code>ex:hatSmiley</code>	<code>rdfs:range</code>	<code>rdf:Literal .</code>
<code>ex:böseBemerkung</code>	<code>ex:hatSmiley</code>	<code>">:->"^^XMLLiteral .</code>

Darin wird einerseits ausgesagt, dass als Werte der `hatSmiley`-Property nur Literalwerte (d.h. Elemente von LV) zulässig sind, andererseits wird dieser Property in einem konkreten Tripel ein (aufgrund der Zeichen „>“ ohne öffnende Gegenstücke) nicht wohlgeformtes Literal zugeordnet, welches definitionsgemäß *nicht* als Literalwert interpretiert werden darf. Folglich kann es keine RDFS-Interpretation geben, die beide Tripel als *wahr* auswertet.